Binary Polar Code Kernels from Code Decompositions

Noam Presman, Ofer Shapira and Simon Litsyn
School of Electrical Engineering, Tel Aviv University, Ramat Aviv 69978 Israel.
e-mails: {presmann, ofershap, litsyn}@eng.tau.ac.il.

Abstract

Code decompositions (a.k.a code nestings) are used to design good binary polar code kernels. The proposed kernels are in general non-linear and show a better rate of polarization under *successive cancelation* decoding, than the ones suggested by Korada et al., for the same kernel dimensions. In particular, kernels of sizes 14, 15 and 16 are constructed and shown to provide polarization rates better than any binary kernel of such sizes.

1 Introduction

Polar codes were introduced by Arikan [1] and provided a scheme for achieving the symmetric capacity of binary memoryless channels (B-MC) with polynomial encoding and decoding complexity. Arikan used a simple construction based on the following linear kernel

$$G_2 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}.$$

In this scheme, a $2^n \times 2^n$ matrix, $G_2^{\bigotimes n}$, is generated by performing the Kronecker power on G_2. An input vector $\mathbf{u}$ of length $N = 2^n$ is transformed to an N length vector $\mathbf{x}$ by multiplying a certain permutation of the vector $\mathbf{u}$ by $G_2^{\bigotimes n}$. The vector $\mathbf{x}$ is transmitted through N independent copies of the memoryless channel, W. This results in new N (dependent) channels between the individual components of $\mathbf{u}$ and the outputs of the channels. Arikan showed that these channels exhibit the phenomenon of polarization under successive cancelation decoding. This means that as n grows there is a proportion of $I(W)$ (the symmetric channel capacity) of the channels that become clean channels (i.e. having the capacity approaching 1) and the rest of the channels become completely noisy (i.e. with the capacity approaching 0). An important question is how fast the polarization occurs in terms of the codes' length N. In [2], the rate of polarization was analyzed for the 2×2 kernel, and it was proven that the rate is $O\left(2^{-N^{0.5}}\right)$. More specifically the authors showed that

$$\liminf_{n\to\infty} \Pr\left(Z_n \leq 2^{-N^{\beta}}\right) = I(W) \quad \text{for} \quad \beta < 0.5 \tag{1}$$

$$\liminf_{n\to\infty} \Pr\left(Z_n \geq 2^{-N^{\beta}}\right) = 1 \quad \text{for} \quad \beta > 0.5, \tag{2}$$

where $\{Z_n\}_{n\geq 0}$ is the Bhattacharyya random sequence corresponding to Arikan's random tree process [1].

In [3], Korada *et al.* studied the use of alternatives to G_2 for the symmetric B-MC. They gave sufficient conditions for polarization when linear binary kernels are used over the symmetric B-MC channels. Furthermore, the notion of the rate of polarization was generalized for polar codes based on linear codes having generating matrix G of dimensions $\ell \times \ell$. The rate of polarization was quantified by the exponent of the kernel $E(G)$, which plays the general role of the threshold (equal 0.5) appearing in (1) and (2) (note that here $N = \ell^n$). Korada *et al.* showed that $E(G) \leq 0.5$ for all binary linear kernels of dimension $\ell \leq 15$, which is the kernel exponent found for Arikan's 2×2 kernel, and that for $\ell = 16$ there exists a

code generator matrix G in which $E(G) = 0.51828$, and this is the maximum exponent achievable by a binary linear kernel up to this dimension. Furthermore, for optimal linear kernels, the exponent $E(G)$ approaches 1 as $\ell \to \infty$.

In [4], Mori and Tanaka considered the general case of a mapping $g(\cdot)$, which is not necessarily linear and binary, as a basis for channel polarization constructions. They gave sufficient conditions for polarization and generalized the exponent for these cases. In [5], they considered non-binary, however linear, kernels based on Reed-Solomon codes and Algebraic Geometry codes and showed that their exponents are by far better than the exponents of the known binary kernels. This is true even for such a small kernel dimension as $\ell = 4$ and the alphabet size $q = 4$, in which $E\left(G\right) = 0.573120$.

In this paper, we propose designing good binary kernels (in the sense of large exponent), by using code decompositions (a.k.a code nestings). The kernels we suggest show better exponents than the ones considered in [3]. Moreover, we describe binary non-linear kernels of sizes 14, 15 and 16 providing a superior polarization exponent than any binary linear kernel.

The paper is organized as follows. In Section 2, we describe building kernels that are related to decompositions of codes into sub-codes. Furthermore, by using results from [4], we observe that the exponent of these kernels is a function of the partial minimum distances between the sub-codes. We then develop in Section 3 an upper-bound on the exponent of dimension ℓ. In Section 4, we give examples of known code decompositions which result in binary kernels that achieve the upper-bounds from Section 3.

2 Preliminaries

We consider kernels that are based on bijective binary transformations. A channel polarization kernel of dimension ℓ, denoted by $g(\cdot)$, is a bijective mapping

$$g : \{0,1\}^{\ell} \to \{0,1\}^{\ell}.$$

This means that $g(\mathbf{u}) = \mathbf{x}, \quad \mathbf{u}, \mathbf{x} \in \{0,1\}^{\ell}$. Denote the output components of the transformation by

$$g_i(\mathbf{u}) = x_i \quad i \in [\ell],$$

where for a natural number ℓ, we denote $[\ell] = \{1, 2, 3, ..., \ell\}$. For $i \geq j$, let $\mathbf{u}_j^i = (u_j, ..., u_i)$ be the sub-vector of $\mathbf{u}$ of length $i-j+1$ (if $i < j$ we say that $\mathbf{u}_j^i = ()$, the empty vector, and its length is 0). It is convenient to denote by $g^{(\mathbf{v}_1^i)} : \{0,1\}^{\ell-i} \to \{0,1\}^{\ell}$, the restriction of $g(\cdot)$ to the set $\left\{\mathbf{v}_1^i \mathbf{u}_1^{\ell-i} | \mathbf{u}_1^{\ell-i} \in \{0,1\}^{\ell-i}\right\}$, that is

$$g^{(\mathbf{v}_1^i)}(\mathbf{u}_1^{\ell-i}) = g(\mathbf{v}_1^i \mathbf{u}_1^{\ell-i}) \quad i \in [\ell-1].$$

Next, we consider code decompositions. The initial code is partitioned to several sub-codes having the same size. Each of these sub-codes can be further partitioned. Here we choose as the initial code, the total space of length ℓ binary vectors, and denote it by $T_1^{()} = \{0,1\}^{\ell}$. This set is partitioned to m_1 equally sized sub-codes $T_2^{(0)}, T_2^{(1)}, ..., T_2^{(m_1-1)}$, and each sub-code $T_2^{(b_1)}$ is in turn partitioned to m_2 equally sized codes $T_3^{(b_1,0)}, T_3^{(b_1,1)}, ..., T_3^{(b_1,m_2-1)}$ ($b_1 \in \{0, 1, ..., m_1 - 1\}$). This partitioning may be further carried on.

Definition 1 *The set $\{T_1, ..., T_m\}$ is called a decomposition of $\{0,1\}^{\ell}$, if $T_1^{()} = \{0,1\}^{\ell}$, and $T_i^{(\mathbf{b}_1^{i-1})}$ is partitioned into m_i equally sized sets $\left\{T_{i+1}^{(\mathbf{b}_1^{i-1} b_i)}\right\}_{b_i=0,1,...,m_i-1}$, of size $\frac{2^{\ell}}{\prod_{j=1}^{i} m_j}$ ($i \in [m-1]$). We denote the set of sub-codes of level number i by*

$$T_i = \left\{T_i^{(\mathbf{b}_1^{i-1})} | b_j \in \{0,1,2,...,m_j-1\}, j \in [i-1]\right\}.$$

The partition is usually described by the following chain of codes parameters

$$(n_1, k_1, d_1) - (n_2, k_2, d_2) - ... - (n_m, k_m, d_m),$$

if for each $\mathcal{T} \in T_i$ we have that $\mathcal{T}$ is a code of length n_i, size 2^{k_i} and minimum distance at least d_i.

If the sub-codes of the decompositions are cosets, then we say that $\{T_1, ..., T_m\}$ is a decomposition into cosets. In this case, for each T_i the sub-code that contains the zero codeword is called the representative sub-code, and a minimal weight codeword for each coset is called the coset leader. If all the sub-codes in the decomposition are cosets of linear codes, we say that the decomposition is linear.

Example 1 *As an example consider $\ell = 4$ and the 4×4 binary matrix*

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

A partition into cosets, having the following chain of parameters $(4, 4, 1) - (4, 3, 2) - (4, 1, 4)$, can be implied by the matrix. This is done by taking $T_1^{()} = \{0, 1\}^4$, which is partitioned to the even weight codewords and odd weight codewords cosets, i.e. $T_2^{(0)} = \left\{\mathbf{u}_1^4 | \sum_{i=1}^4 u_i \equiv 0 (\bmod 2)\right\}$, $T_2^{(1)} = \left\{\mathbf{u}_1^4 | \sum_{i=1}^4 u_i \equiv 1 (\bmod 2)\right\}$, these cosets are in turn partitioned to anti podalic pairs, $T_3^{(0,0)} = \{0000, 1111\}$, $T_3^{(0,1)} = \{1010, 0101\}$, $T_3^{(0,2)} = \{1100, 0011\}$, $T_3^{(0,3)} = \{0110, 1001\}$, and $T_3^{(1,b)} = [1000] + T_3^{(0,b)}$ ($b \in \{0, 1, 2, 3\}$). Note, that in order to describe this partition, it suffices to describe the representatives and the coset leaders for the partition of the representatives.

A binary transformation can be associated to a code decomposition in the following way.

Definition 2 *Let $\{T_1, T_2, ..., T_{\ell+1}\}$ be a code decomposition of $\{0, 1\}^\ell$, such that $m_i = 2$ for each $i \in [\ell]$. Note that the code $T_i^{(\mathbf{b}_1^{i-1})}$ is of size $2^{\ell-i+1}$, and specifically $T_{\ell+1}^{(b_1, b_2, ..., b_\ell)}$ contains only one codeword. We call such a decomposition a binary decomposition. The transformation $g(\cdot) : \{0, 1\}^\ell \to \{0, 1\}^\ell$ induced by this binary code decomposition is defined as follows.*

$$g(\mathbf{u}_1^\ell) = \mathbf{x}_1^\ell \quad \textit{if } \mathbf{x}_1^\ell \in T_{\ell+1}^{(\mathbf{u}_1^\ell)}. \tag{3}$$

Following the definition, we can observe, that a sequential decision making on the bits of the input to the transformation ($\mathbf{u}_1^\ell$) given a noisy observation of the output is actually a decision on the sub-code to which the transmitted vector belongs to. As such, deciding on the first bit u_1 is actually deciding if the transmitted vector belongs to $T_2^{(0)}$ or to $T_2^{(1)}$. Once we decided on u_1, we assume that we transmitted a codeword of $T_2^{(u_1)}$ and by deciding on u_2 we choose the appropriate refinement or sub-code of $T_2^{(u_1)}$, i.e. we should decide between the candidates $T_3^{(u_1,0)}$ and $T_3^{(u_1,1)}$. Due to this fact, it comes as no surprise that the Hamming distances between two candidate sub-codes plays an important role when considering the rate of polarization.

Definition 3 *For a binary code decomposition as in Definition 2, the Hamming distances between sub-codes in the decomposition are defined as follows:*

$$D_{min}^{(i)}(\mathbf{u}_1^{i-1}) = \min\left\{d_H(\mathbf{c}_1, \mathbf{c}_2)\middle|\mathbf{c}_1 \in T_{i+1}^{(\mathbf{u}_1^{i-1} \cdot 0)}, \mathbf{c}_2 \in T_{i+1}^{(\mathbf{u}_1^{i-1} \cdot 1)}\right\},$$

$$D_{min}^{(i)} = \min\left\{D_{min}^{(i)}(\mathbf{u}_1^{i-1}) | \mathbf{u}_1^{i-1} \in \{0, 1\}^{i-1}\right\}.$$

A transformation $g(\cdot)$ can be used as a building block for a recursive construction of a transformation of greater length, in a similar manner to [1]. We specify this construction explicitly in the next definition.

Definition 4 *Given a transformation $g(\cdot)$ of dimension ℓ, we construct a mapping $g^{(m)}(\cdot)$ of dimension ℓ^m (i.e. $g^{(m)}(\cdot) : \{0,1\}^{\ell^m} \to \{0,1\}^{\ell^m}$) in the following recursive fashion.*

$$g^{(1)}(\mathbf{u}_1^\ell) = g(\mathbf{u}_1^\ell) \;;$$

$$\begin{aligned} g^{(m)} = \Big[& g^{(m-1)}\left(\gamma_{1,1}, \gamma_{2,1}, \gamma_{3,1}, \ldots, \gamma_{\ell^{m-1},1}\right), \\ & g^{(m-1)}\left(\gamma_{1,2}, \gamma_{2,2}, \gamma_{3,2}, \ldots, \gamma_{\ell^{m-1},2}\right), \ldots, \\ & g^{(m-1)}\left(\gamma_{1,\ell}, \gamma_{2,\ell}, \gamma_{3,\ell}, \ldots, \gamma_{\ell^{m-1},\ell}\right) \Big], \end{aligned}$$

where

$$\gamma_{i,j} = g_j\left(\mathbf{u}_{(i-1)\cdot\ell+1}^{i\cdot\ell}\right) \quad 1 \le i \le \ell^{m-1} \quad 1 \le j \le \ell.$$

The transformation $g^{(m)}(\cdot)$ can be used to transmit data over the B-MC channel. The method of successive cancelation can now be used to decode, with decoding complexity of $O\left(2^\ell \cdot N \cdot \log_\ell(N)\right)$ as in [1].

We use the same channel definition, the corresponding symmetric capacity and the Bhattacharyya parameter as in [1, 3, 4]. Note that for uniform binary random vectors U_1^ℓ, and $X_1^\ell = g\left(U_1^\ell\right)$ we have that $I(Y_1^\ell; U_1^\ell) = I(Y_1^\ell; X_1^\ell)$, because the transformation $g(\cdot)$ is invertible. Furthermore, since we consider memoryless channels, we have $I(Y_1^\ell; X_1^\ell) = \ell \cdot I(Y_1; X_1) = \ell \cdot I(W)$, and on the other hand

$$I(Y_1^\ell; U_1^\ell) = \sum_{i=1}^{\ell} I(Y_1^\ell; U_i | U_1^{i-1}) = \sum_{i=1}^{\ell} I(W^{(i)}).$$

Define the tree process of the channels generated by the kernels, in the same way as it was done in [1] and generalized in [3]. A random sequence $\{W_n\}_{n\ge 0}$ is defined such that $W_n \in \left\{W^{(i)}\right\}_{i=1}^{\ell^n}$ with

$$W_0 = W$$

$$W_{n+1} = W_n^{(B_{n+1})},$$

where $\{B_n\}_{n\ge 1}$ is a sequence of i.i.d random variables uniformly distributed over the set $\{0, 1, 2, ..., \ell - 1\}$. In a similar manner, the symmetric capacity corresponding to the channels $\{I_n\}_{n\ge 0} = \{I(W_n)\}_{n\ge 0}$ and the Bhattacharyya parameters random variables $\{Z_n\}_{n\ge 0} = \{Z(W_n)\}_{n\ge 0}$ are defined. Just as in [1, Proposition 8], we can prove that the random sequence $\{I_n\}_{n\ge 0}$ is a bounded martingale, and it is uniform integrable which means it converges almost surely to I_∞ and that $\mathbb{E}\{I_\infty\} = I(W)$. Now, if we can show that $Z_n \to Z_\infty$ w.h.p such that $Z_\infty \in \{0, 1\}$, by the relations between the channel's information and the Bhattacharyya parameter [1, Proposition 1], we have that $I_\infty \in \{0, 1\}$. But, this means that $\Pr(I_\infty = 1) = \mathbb{E}\{I_\infty\} = I(W)$, which is the channel polarization phenomenon.

Proposition 1 *Let $g(\cdot)$ be a binary transformation of dimension ℓ, induced by a binary code decomposition $\{T_1, T_2, ..., T_{\ell+1}\}$. If there exists $\mathbf{u}_1^{\ell-1} \in \{0,1\}^{\ell-1}$ such that $D_{min}^{(\ell)}(\mathbf{u}_1^{\ell-1}) \ge 2$, then $\Pr(I_\infty = 1) = I(W)$.*

Proof In [4, Corollary 11], sufficient conditions are given for

$$\lim_{n\to\infty} \Pr\left(Z_n \in (\delta, 1-\delta)\right) = 0 \quad \forall \delta \in (0, 0.5). \tag{4}$$

The first condition is that there exists a vector $\mathbf{u}_1^{\ell-1}$, indices $i, j \in [\ell]$ and permutations $\sigma(\cdot)$, and $\tau(\cdot)$ on $\{0, 1\}$ such that

$$g_i^{(\mathbf{u}_1^{\ell-1})}(u_\ell) = \sigma(u_\ell) \quad \text{and} \quad g_j^{(\mathbf{u}_1^{\ell-1})}(u_\ell) = \mu(u_\ell).$$

This requirement applies here, because if there exists $\mathbf{u}_1^{\ell-1} \in \{0,1\}^{\ell-1}$ such that $D_{min}^{(\ell)}(\mathbf{u}_1^{\ell-1}) \ge 2$, then the two codewords of the code $T_\ell^{(\mathbf{u}_1^{\ell-1})}$, $\mathbf{c}_1$ and $\mathbf{c}_2$, are at Hamming distance at least 2. This means that

there exist at least two indices i, j such that $c_{1,i} \neq c_{2,i}$ and $c_{1,j} \neq c_{2,j}$, therefore $g_i^{(\mathbf{u}_1^{\ell-1})}(u_\ell)$ and $g_j^{(\mathbf{u}_1^{\ell-1})}(u_\ell)$ are both permutations. The second condition is that for any $\mathbf{v}_1^{\ell-1} \in \{0,1\}^{\ell-1}$ there exist an index $m \in [\ell]$ and a permutation $\mu(\cdot)$ on $\{0,1\}$ such that

$$g_m^{(\mathbf{v}_1^{\ell-1})}(v_\ell) = \mu(v_\ell).$$

This requirement also applies here, by noting that for each $\mathbf{v}_1^{\ell-1} \in \{0,1\}^{\ell-1}$ the two codewords of the set $T_\ell^{(\mathbf{v}_1^{\ell-1})}$ are at Hamming distance at least 1. This means that (4) holds, which implies that $I_\infty \in \{0,1\}$ almost surely, and therefore $\Pr(I_\infty = 1) = I(W)$. $\diamondsuit$

The next proposition on the rate of polarization is an easy consequence of [4, Theorem 19] and Proposition 1.

Proposition 2 *Let $g(\cdot)$ be a bijective transformation of dimension ℓ, induced by code partitioning $\{T_1, T_2, ..., T_{\ell+1}\}$. If there exists $\mathbf{u}_1^{\ell-1} \in \{0,1\}^{\ell-1}$ such that $D_{min}^{(\ell)}(\mathbf{u}_1^{\ell-1}) \geq 2$, then*

(i) For any $\beta < E(g)$

$$\lim_{n\to\infty} \Pr\left(Z_n \leq 2^{-\ell^{n\beta}}\right) = I(W),$$

(ii) For any $\beta > E(g)$

$$\lim_{n\to\infty} \Pr\left(Z_n \geq 2^{-\ell^{n\beta}}\right) = 1,$$

where $E(g) = \frac{1}{\ell}\sum_{i=1}^{\ell} \log_\ell\left(D_{min}^{(i)}\right)$.

Naturally, we would like to find kernels maximizing $E(g)$. In the next section we consider upper bounds on the maximum achievable exponent per dimension ℓ.

3 Bounds on the Optimal Exponent

We define the optimal exponent per dimension ℓ as

$$E_\ell = \max_{g:\{0,1\}^\ell \to \{0,1\}^\ell} E(g). \tag{5}$$

Note that in [3], E_ℓ was defined as a maximization over the set of binary linear kernels, and here we extend the definition for general kernels. Furthermore, a lower bound on the kernel using Gilbert-Vershamov technique also applies in this case [3, Lemma 20]. The following lemma is a generalization of [3, Lemma 18].

Lemma 1 (Generalization of [3], Lemma 20) *Let $g : \{0,1\}^\ell \to \{0,1\}^\ell$ be a polarizing kernel. Fix $k \in [\ell-1]$ and define a mapping*

$$\tilde{g}\left(\mathbf{v}_1^\ell\right) = g\left(\mathbf{v}_1^{k-1}\ , v_{k+1}\ , v_k\ , \mathbf{v}_{k+2}^\ell\right), \tag{6}$$

i.e in this mapping the coordinates k and $k+1$ are swapped. Let $\left\{D_{min}^{(i)}\right\}_{i=1}^{\ell}$ and $\left\{\tilde{D}_{min}^{(i)}\right\}_{i=1}^{\ell}$ denote the partial distances of $g(\cdot)$ and $\tilde{g}(\cdot)$ respectively. If $D_{min}^{(k)} > D_{min}^{(k+1)}$ then

(i) $E(g) \leq E(\tilde{g})$

(ii) $\tilde{D}_{min}^{(k)} < \tilde{D}_{min}^{(k+1)}$

Proof We follow the path of the proof of [3, Lemma 20]. It will be useful to introduce the following equivalent definition of the partial distance sequence.

$$D_{\min}^{(i)} = \min\left\{d_H\left(g\left(\mathbf{w}_1^{i-1}, 0, \mathbf{u}_{i+1}^{\ell}\right), g\left(\mathbf{w}_1^{i-1}, 1, \mathbf{v}_{i+1}^{\ell}\right)\right) \Big| \mathbf{w}_1^{i-1}, \mathbf{u}_{i+1}^{\ell}, \mathbf{v}_{i+1}^{\ell}\right\} \tag{7}$$

According to this definition it is easy to see that

$$D_{\min}^{(i)} = \tilde{D}_{\min}^{(i)} \quad i \in [\ell]\backslash\{k, k+1\}. \tag{8}$$

Hence, it suffices to show that

$$D_{\min}^{(k)} \cdot D_{\min}^{(k+1)} \leq \tilde{D}_{\min}^{(k)} \cdot \tilde{D}_{\min}^{(k+1)} \tag{9}$$

in order to prove (i).

Using (7), we have

$$D_{\min}^{(k)} = \min\left\{d_H\left(g\left(\mathbf{w}_1^{k-1}, 0, \mathbf{u}_{k+1}^{\ell}\right), g\left(\mathbf{w}_1^{k-1}, 1, \mathbf{v}_{k+1}^{\ell}\right)\right) \Big| \mathbf{w}_1^{k-1}, \mathbf{u}_{k+1}^{\ell}, \mathbf{v}_{k+1}^{\ell}\right\} \tag{10}$$

$$\tilde{D}_{\min}^{(k)} = \min\left\{d_H\left(g\left(\mathbf{w}_1^{k-1}, u_k, 0, \mathbf{u}_{k+2}^{\ell}\right), g\left(\mathbf{w}_1^{k-1}, v_k, 1, \mathbf{v}_{k+2}^{\ell}\right)\right) \Big| \mathbf{w}_1^{k-1}, \mathbf{u}_{k+2}^{\ell}, \mathbf{v}_{k+2}^{\ell}, u_k, v_k\right\} \tag{11}$$

$$D_{\min}^{(k+1)} = \min\left\{d_H\left(g\left(\mathbf{w}_1^{k}, 0, \mathbf{u}_{k+2}^{\ell}\right), g\left(\mathbf{w}_1^{k}, 1, \mathbf{v}_{k+2}^{\ell}\right)\right) \Big| \mathbf{w}_1^{k}, \mathbf{u}_{k+2}^{\ell}, \mathbf{v}_{k+2}^{\ell}\right\}. \tag{12}$$

$$\tilde{D}_{\min}^{(k+1)} = \min\left\{d_H\left(g\left(\mathbf{w}_1^{k-1}, 0, w_{k+1}, \mathbf{u}_{k+2}^{\ell}\right), g\left(\mathbf{w}_1^{k-1}, 1, w_{k+1}, \mathbf{v}_{k+2}^{\ell}\right)\right) \Big| \mathbf{w}_1^{k-1}, w_{k+1}, \mathbf{u}_{k+2}^{\ell}, \mathbf{v}_{k+2}^{\ell}\right\}. \tag{13}$$

Because the set on which we perform the minimization in (13) is a subset of the set on which we preform the minimization in (10) we have that $D_{\min}^{(k)} \leq \tilde{D}_{\min}^{(k+1)}$. On the other hand, the minimization in (11) can be expressed as $\tilde{D}_{\min}^{(k)} = \min\left\{\Delta_1, \Delta_2\right\}$, where

$$\Delta_1 = \min\left\{d_H\left(g\left(\mathbf{w}_1^{k-1}, w_k, 0, \mathbf{u}_{k+2}^{\ell}\right), g\left(\mathbf{w}_1^{k-1}, w_k, 1, \mathbf{v}_{k+2}^{\ell}\right)\right) \Big| \mathbf{w}_1^{k-1}, \mathbf{u}_{k+2}^{\ell}, \mathbf{v}_{k+2}^{\ell}, w_k\right\} \tag{14}$$

$$\Delta_2 = \min\left\{d_H\left(g\left(\mathbf{w}_1^{k-1}, w_k, 0, \mathbf{u}_{k+2}^{\ell}\right), g\left(\mathbf{w}_1^{k-1}, 1-w_k, 1, \mathbf{v}_{k+2}^{\ell}\right)\right) \Big| \mathbf{w}_1^{k-1}, \mathbf{u}_{k+2}^{\ell}, \mathbf{v}_{k+2}^{\ell}, w_k\right\}. \tag{15}$$

We see that $\Delta_1 = D_{\min}^{(k+1)}$ and $\Delta_2 \geq D_{\min}^{(k)}$. So, $\tilde{D}_{\min}^{(k)} = D_{\min}^{(k+1)}$, because $D_{\min}^{(k)} > D_{\min}^{(k+1)}$. So this proves (9) and therefore (i). Now,

$$\tilde{D}_{\min}^{(k)} = D_{\min}^{(k+1)} < D_{\min}^{(k)} \leq \tilde{D}_{\min}^{(k+1)},$$

which results in (ii). $\diamondsuit$

Lemma 1 implies that when seeking the optimal exponent,E_ℓ, for a given dimension ℓ, it suffices to consider kernels with non-decreasing partial distance sequences. This observation also results in [3, Lemma 22]

Lemma 2 ([3],Lemma 22) *Let $d(n,k)$ denote the largest possible minimum distance of a binary code of length n and size 2^k. Then,*

$$E_\ell \leq \frac{1}{\ell}\sum_{i=1}^{\ell} \log_\ell\left(d(\ell, \ell-i+1)\right) \tag{16}$$

Proof Consider a polarizing kernel $g(\cdot)$ having partial distance sequence $\left\{D_{\min}^{(i)}\right\}_{i=1}^{\ell}$. Because of Lemma 1, we can assume that the sequence is non decreasing (otherwise, we can find a kernel that is having a non-decreasing sequence with at least the same exponent). Note that

$$D_{\min}^{(k)} = \min_{i\geq k} D_{\min}^{(i)} = \min_{\mathbf{u}_1^{k-1}}\left\{\min\left\{d_H(\mathbf{c}_1, \mathbf{c}_2)\Big| \mathbf{c}_1, \mathbf{c}_2 \in T_k^{\left(\mathbf{u}_1^{k-1}\right)}, \mathbf{c}_1 \neq \mathbf{c}_2\right\}\right\} \leq d(\ell, \ell-k+1), \tag{17}$$

where the second inequality is due to the fact that each of the codes in the inner minimum, (i.e. $T_k^{(\mathbf{u}_1^{k-1})}$), is of size $2^{\ell-k+1}$ and length ℓ. ♢

As already noted in [3], the shortcoming of (16) as an upper-bound, is that the dependencies between the partial distances are not exploited. For binary and linear kernels, [3, Lemma 26] gives an improved upper bound utilizing these dependencies. In the sequel we develop an upper bound that is applicable to general kernels. The basic idea of the bound we develop, is to express the partial distance sequence of a kernel, in terms of distance distributions of a code.

For a code $\mathcal{C}$ of length ℓ and size M we define the distance distribution as

$$B_i = \frac{1}{M}\left|\left\{(\mathbf{c}_1, \mathbf{c}_2)\middle| d_H(\mathbf{c}_1, \mathbf{c}_2) = i\right\}\right| \quad 0 \leq i \leq \ell. \tag{18}$$

Note that $B_0 = 1$ and

$$\sum_{i=1}^{\ell} B_i = M - 1. \tag{19}$$

Now, given a non decreasing partial distance sequence $\left\{D_{\min}^{(i)}\right\}_{i=1}^{\ell}$ we choose an arbitrary $k \in [\ell]$ and consider the sub-sequence $\left\{D_{\min}^{(i)}\right\}_{i=k}^{\ell}$. Using the reasoning that led to (17), we observe that we need to consider the sub-codes $\left\{T_k^{(\mathbf{u}_1^{k-1})}\right\}_{\mathbf{u}_1^{k-1}\in\{0,1\}^{k-1}}$ of size $M = 2^{\ell-k+1}$, but whereas in (16) we considered only the minimum distance, here we may have additional requirements from the distance distribution of the code. Let's begin by understanding the meaning of $D_{\min}^{(\ell)}$ (the last element of the sequence). By definition, the code $T_k^{(\mathbf{u}_1^{k-1})}$ is decomposed into $\frac{2^{\ell-k+1}}{2}$ sub-codes of size 2, such that in each one the distance between the 2 codewords is at least $D_{\min}^{(\ell)}$. This means that we must fulfill the following requirement

$$\sum_{i=D_{\min}^{(\ell)}}^{\ell} B_i \geq 1, \tag{20}$$

where $\{B_o\}_{i=0}^{\ell}$ is the distance distribution of $T_k^{(\mathbf{u}_1^{k-1})}$. Now, let's proceed to $D_{\min}^{(\ell-1)}$. This item implies that there are $\frac{2^{\ell-k+1}}{2^2}$ sub-codes of $T_k^{(\mathbf{u}_1^{k-1})}$ of 4 codewords that each one of them can be decomposed into 2 sub-codes of 2 code-words having minimum distance between the sub-codes of at least $D_{\min}^{(\ell-1)}$. From this, we deduce that there are $2 \cdot 2^{\ell-k+1}$ pairs of codewords having their distance at least $D_{\min}^{(\ell-1)}$. These pairs are an addition to the the ones we counted in (20). Thus, because we assume that the partial distance sequence is non-decreasing, we have the following requirement.

$$\sum_{i=D_{\min}^{(\ell-1)}}^{\ell} B_i \geq 3. \tag{21}$$

Note that if $D_{\min}^{(\ell-1)} = D_{\min}^{(\ell)}$ then (20) is redundant given (21). In the general case, when considering $D_{\min}^{(\ell-r)}$, where $0 \leq r \leq \ell - k$, we take into account $\frac{2^{\ell-k+1}}{2^{r+1}}$ sub-codes of $T_k^{(\mathbf{u}_1^{k-1})}$, each one of size 2^{r+1} and each one can be partitioned into two sub-codes of which the minimum distance between them is $D_{\min}^{(\ell-r)}$. So, there are $2 \cdot \frac{2^{\ell-k+1}}{2^{r+1}} \cdot (2^r)^2 = M \cdot 2^r$ codewords pairs (that were not counted at the previous steps) such that their distance is at least $D_{\min}^{(\ell-r)}$. Summarizing, we get the following set of $\ell - k + 1$ inequalities

$$\sum_{i=D_{\min}^{(\ell-r)}}^{\ell} B_i \geq \sum_{j=0}^{r} 2^j = 2^{r+1} - 1 \qquad 0 \leq r \leq \ell - k. \tag{22}$$

By Delsarte [6], we can specify additional linear requirements on the distance distribution, by

$$\sum_{j=1}^{\ell} B_j \cdot P_i(j) \geq -\binom{\ell}{i} \qquad 0 \leq i \leq \ell, \tag{23}$$

where $P_k(x)$ is the Krawtchouk polynomial, which is defined as

$$P_k(x) = \sum_{m=0}^{k} (-1)^m \binom{x}{m}\binom{\ell - x}{k-m}. \tag{24}$$

In addition, the following is also an obvious requirement

$$B_i \geq 0 \quad i \in [\ell]. \tag{25}$$

We see that requirements (19),(22),(23) and (25) are all linear. A partial distance sequence that corresponds to a kernel must be able to fulfill these requirements for every $k \in [\ell]$. So, taking the maximum exponent corresponding to a partial distance sequence that fulfils the requirement for each $k \in [\ell]$ results in an upper-bound on the exponent. Checking the validity of a sequence can be done by linear programming methods (we need to check if the polytope is not empty). We now turn to give two simple examples of the method, and after them we present a variation on this development that leads to a stronger bound.

Example 2 *Consider $\ell = 3$. Let $\left\{D_{min}^{(i)}\right\}_{i=1}^{3}$ be the partial distance sequence. Note first that by the singelton bound $D_{min}^{(k)} \leq k$. We first consider the possibility that $D_{min}^{(3)} = 3$ and $D_{min}^{(2)} = 2$. This assumption is translated by (19) and (22) to*

$$B_2 + B_3 = 3 \qquad , B_3 \geq 1 \quad , \quad B_2, B_3 \geq 0 \tag{26}$$

By (23) for $i = 1$ we have

$$B_2 \cdot P_1(2) + B_3 \cdot P_1(3) \geq -3$$

$$-B_2 - 3 \cdot B_3 \geq -3 \quad \Longrightarrow_{(26)} B_2 = 0, B_3 = 3 \tag{27}$$

on the other hand for (23) $i = 3$ we have

$$B_2 - B_3 \geq -1$$

which is a contradiction to (27). The next best candidate is a sequence having $D_{min}^{(2)} = D_{min}^{(3)} = 2$, this sequence is feasible by considering the following generating matrix

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}.$$

This proves that $E_3 = \frac{1}{3}\log_3 4 \approx 0.42062$.

Example 3 *Consider $\ell = 4$. Let $\left\{D_{min}^{(i)}\right\}_{i=1}^{4}$ be the partial distance sequence. We first consider the possibility that $D_{min}^{(4)} = D_{min}^{(3)} = 3$ (if this possibility is eliminated it means that $D_{min}^{(4)} = 4, D_{min}^{(3)} = 3$ is also not possible). (19) and (22) are translated to*

$$B_3 + B_4 = 3 \qquad B_3, B_4 \geq 0 \tag{28}$$

By (23) for $i=1$ we have

$$B_3 \cdot P_1(3) + B_4 \cdot P_1(4) \geq -4$$

$$-2 \cdot B_3 - 4 \cdot B_4 \geq -4 \Longrightarrow_{(28)} B_3 + 2(3 - B_3) \leq 2 \Longrightarrow B_3 \geq 4$$

which is a contradiction to (28). The next best candidate is

$$D_{min}^{(4)} = 4, D_{min}^{(3)} = 2, D_{min}^{(2)} = 2, D_{min}^{(1)} = 1,$$

which can be achieved by a binary linear kernel induced by the generating matrix

$$\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}^{\otimes 2}.$$

This proves that $E_4 = 0.5$.

The idea of transforming the partial distance sequence into requirements on distance distributions can be further refined. As we did before, we begin our discussion by considering the sub-sequence $\left\{D_{\min}^{(i)}\right\}_{i=k}^{\ell}$. We start by giving an interpretation to $D_{\min}^{(\ell)}$ (the last element of the sequence). By definition, the code $T_k^{\left(\mathbf{u}_1^{k-1}\right)}$ is decomposed into $\frac{2^{\ell-k+1}}{2}$ sub-codes of size 2, where in each one the distance between the 2 codewords is at least $D_{\min}^{(\ell)}$. Denote by $B_i^{\left(\mathbf{u}_1^{\ell-1}\right)}$ $i \in [\ell]$ the partial distance distribution of the sub-code $T_k^{\left(\mathbf{u}_1^{\ell-1}\right)}$ of the code $T_k^{\left(\mathbf{u}_1^{k-1}\right)}$. By definition we have

$$B_i^{\left(\mathbf{u}_1^{\ell-1}\right)} = \frac{1}{2}\left|\left\{d_H(\mathbf{c}_1, \mathbf{c}_2) = i \middle| \mathbf{c}_1, \mathbf{c}_2 \in T_\ell^{\left(\mathbf{u}_1^{\ell-1}\right)}\right\}\right|. \tag{29}$$

Obviously,

$$\sum_{i=D_{\min}^{(\ell)}}^{\ell} B_i^{\left(\mathbf{u}_1^{\ell-1}\right)} = 1 \quad \forall \mathbf{u}_k^{\ell-1} \in \{0,1\}^{\ell-k}, \tag{30}$$

$$\sum_{j=1}^{\ell} B_j^{\left(\mathbf{u}_1^{\ell-1}\right)} \cdot P_i(j) \geq -\binom{\ell}{i} \qquad 0 \leq i \leq \ell, \forall \mathbf{u}_k^{\ell-1} \in \{0,1\}^{\ell-k}. \tag{31}$$

Define the average of this distribution over all the sub-codes of $T_k^{\left(\mathbf{u}_1^{k-1}\right)}$, i.e.

$$\bar{B}_i^{(\ell)} = \frac{1}{2^{\ell-k}} \sum_{\mathbf{u}_k^{\ell-1} \in \{0,1\}^{\ell-k}} B_i^{\left(\mathbf{u}_1^{\ell-1}\right)} \quad i \in [\ell]. \tag{32}$$

Note that

$$\bar{B}_i^{(\ell)} = \frac{1}{M}\left|\left\{d_H(\mathbf{c}_1, \mathbf{c}_2) = i \middle| \mathbf{c}_1, \mathbf{c}_2 \in T_k^{\left(\mathbf{u}_1^{\ell-1}\right)}, \mathbf{u}_k^{\ell-1} \in \{0,1\}^{\ell-k}\right\}\right| \tag{33}$$

and

$$\sum_{i=D_{\min}^{(\ell)}}^{\ell} \bar{B}_i^{(\ell)} = 1, \tag{34}$$

$$\sum_{j=1}^{\ell} \bar{B}_j^{(\ell)} \cdot P_i(j) \geq -\binom{\ell}{i} \qquad 0 \leq i \leq \ell. \tag{35}$$

Let's proceed to $D_{\min}^{(\ell-1)}$. By definition, the code $T_k^{(\mathbf{u}_1^{k-1})}$ is decomposed into $\frac{2^{\ell-k+1}}{4}$ sub-codes of size 4, where in each one the distance between the 2 codewords is at least $D_{\min}^{(\ell-1)}$. Denote by $B_i^{(\mathbf{u}_1^{\ell-2})}$ $i \in [\ell]$, the distance distribution of the sub-code $T_k^{(\mathbf{u}_1^{\ell-2})}$ of the code $T_k^{(\mathbf{u}_1^{k-1})}$.

$$B_i^{(\mathbf{u}_1^{\ell-2})} = \frac{1}{4}\left|\left\{d_H(\mathbf{c}_1, \mathbf{c}_2) = i \middle| \mathbf{c}_1, \mathbf{c}_2 \in T_{\ell-1}^{(\mathbf{u}_1^{\ell-2})}\right\}\right|. \tag{36}$$

Note that

$$B_i^{(\mathbf{u}_1^{\ell-2})} \geq \frac{1}{2}\left(B_i^{(\mathbf{u}_1^{\ell-2}\cdot 0)} + B_i^{(\mathbf{u}_1^{\ell-2}\cdot 1)}\right). \tag{37}$$

So by introducing the average distance distribution

$$\bar{B}_i^{(\ell-1)} = \frac{1}{2^{\ell-k-1}} \sum_{\mathbf{u}_k^{\ell-2} \in \{0,1\}^{\ell-k-1}} B_i^{(\mathbf{u}_1^{\ell-2})} \quad i \in [\ell], \tag{38}$$

we get

$$\sum_{i=D_{\min}^{(\ell-1)}}^{\ell} \bar{B}_i^{(\ell-1)} = 3, \tag{39}$$

$$\sum_{j=1}^{\ell} \bar{B}_j^{(\ell-1)} \cdot P_i(j) \geq -\binom{\ell}{i} \quad 0 \leq i \leq \ell. \tag{40}$$

and

$$\bar{B}_i^{(\ell-1)} - \bar{B}_i^{(\ell)} \geq 0 \quad 0 \leq i \leq \ell. \tag{41}$$

In the general case, when taking $D_{\min}^{(\ell-r)}$ into account, where $0 \leq r \leq \ell - k$, we essentially consider the $\frac{2^{\ell-k+1}}{2^{r+1}}$ sub-codes of $T_k^{(\mathbf{u}_1^{k-1})}$, each one of size 2^{r+1} and each one can be partitioned into two sub-codes of size 2^r of which the minimum distance between them is $D_{\min}^{(\ell-r)}$. Denote the distance distribution of the sub-code $T_{\ell-r}^{(\mathbf{u}_1^{\ell-(r+1)})}$ as $\left\{B_i^{(\mathbf{u}_1^{\ell-(r+1)})}\right\}_{i\in[\ell]}$ and the average distance distribution as $\left\{\bar{B}_i^{(\ell-r)}\right\}_{i\in[\ell]}$. We have

$$B_i^{(\mathbf{u}_1^{\ell-(r+1)})} = \frac{1}{2^{r+1}}\left|\left\{d_H(\mathbf{c}_1, \mathbf{c}_2) = i \middle| \mathbf{c}_1, \mathbf{c}_2 \in T_{\ell-r}^{(\mathbf{u}_1^{\ell-(r+1)})}\right\}\right|, \tag{42}$$

$$\bar{B}_i^{(\ell-r)} = \frac{1}{2^{\ell-k-r}} \sum_{\mathbf{u}_k^{\ell-(r-1)} \in \{0,1\}^{\ell-k-r}} B_i^{(\mathbf{u}_1^{\ell-(r+1)})} \quad , i \in [\ell], \tag{43}$$

which results in

$$\sum_{i=D_{\min}^{(\ell-r)}}^{\ell} \bar{B}_i^{(\ell-r)} = \sum_{j=0}^{r} 2^j = 2^{r+1} - 1 \tag{44}$$

$$\bar{B}_i^{(\ell-r)} - \bar{B}_i^{(\ell-r+1)} \geq 0 \tag{45}$$

$$\sum_{j=1}^{\ell} \bar{B}_j^{(\ell-r)} \cdot P_i(j) \geq -\binom{\ell}{i} \quad 0 \leq i \leq \ell. \tag{46}$$

We summarize this development.

#	ℓ	optimal sequence	E_ℓ
1	12	$1, 2, 2, 2, 2, 4, 4, 4, 6, 6, 6, 12$	0.49605
2	13	$1, 2, 2, 2, 2, 4, 4, 4, 6, 6, 6, 8, 10$	0.500498
3	14	$1, 2, 2, 2, 2, 4, 4, 4, 6, 6, 6, 8, 8, 8$	0.50194
4	15	$1, 2, 2, 2, 2, 4, 4, 4, 6, 6, 6, 8, 8, 8, 8$	0.507733
5	16	$1, 2, 2, 2, 2, 4, 4, 4, 6, 6, 6, 8, 8, 8, 8, 16$	0.52742

Table 1: E_ℓ per different dimensions

Definition 5 *Let $\{D_i\}_{i\in[\ell]}$ be a monotone non-increasing sequence of non-negative integral numbers, such that $D_i \leq d(\ell, \ell - i + 1)$. We say that this sequence is ℓ dimension Linear Programming (LP) valid if the polytope defined by the following non negative variables $\left\{\bar{B}_i^{(k)} \middle| 1 \leq k \leq \ell, \quad D_k \leq i \leq \ell\right\}$ is not empty.*

$$\sum_{i=D_{\ell-r}}^{\ell} \bar{B}_i^{(\ell-r)} = \sum_{i=0}^{r} 2^i = 2^{r+1} - 1 \quad 0 \leq r \leq \ell - 1 \tag{47}$$

$$\bar{B}_i^{(\ell-r)} - \bar{B}_i^{(\ell-r+1)} \geq 0 \quad 1 \leq r \leq \ell - 1, \quad D_{\ell-r+1} \leq i \leq \ell \tag{48}$$

$$\sum_{j=D_{\ell-r}}^{\ell} \bar{B}_j^{(\ell-r)} \cdot P_i(j) \geq -\binom{\ell}{i} \qquad 0 \leq i \leq \ell, \quad 0 \leq r \leq \ell - 1 \tag{49}$$

Proposition 3 *If $\left\{D_{min}^{(i)}\right\}_{i\in[\ell]}$ is a partial distance sequence corresponding to some binary ℓ dimension kernel $g(\cdot)$, then $\left\{D_{min}^{(i)}\right\}_{i\in[\ell]}$ is ℓ-dimension LP-valid sequence.*

We denote by $\mathcal{V}_{\mathrm{LP}}^{(\ell)}$ the set of ℓ-dimension *LP*-valid sequences. The following proposition is an easy consequence of Proposition 3.

Proposition 4

$$E_\ell \leq \max_{\{D_k\}_{k\in[\ell]} \in \mathcal{V}_{LP}^{(\ell)}} \frac{1}{\ell} \sum_{i=1}^{\ell} \log_\ell D_i. \tag{50}$$

It should be noted that the method of Proposition 4 can be easily generalized to non-binary kernels using the appropriate (non-binary) Krawtchouk polynomials. We computed the bound for several instances of ℓ by carefully enumerating the sequences in $\mathcal{V}_{\mathrm{LP}}^{(\ell)}$ using Wolfram's *Mathematica* LP-Solver. Table 1 contains the results for $12 \leq \ell \leq 16$. In the next section, we give examples of good kernels, that are derived by utilizing results about known code decompositions, for $14 \leq \ell \leq 16$ that achieve the optimal exponent.

4 Designing Kernels by Known Code Decompositions

As we noticed in Section 2, the exponent, $E(g)$, is influenced by Hamming distances between the subsets in the binary partition $\{T_1, ..., T_{\ell+1}\}$. In this section, we use a particular method for getting good distances by using known decompositions, which are not necessarily binary decompositions. The following observation links between general decompositions and binary decompositions.

Observation 1 *If there exists a code decomposition of $\{0,1\}^\ell$ with the following chain of parameters*

$$(\ell, k_1, d_1) - (\ell, k_2, d_2) - ... - (\ell, k_m, d_m),$$

#	ℓ	chain description	lower bound on $E(g)$
1	16	$(16,1)-(15,2)-(11,4)-(8,6)-(5,8)-(1,16)$	0.52742
2	16	$(16,1)-(15,2)-(11,4)-(7,6)-(5,8)-(1,16)$	0.51828
3	15	$(15,1)-(14,2)-(10,4)-(7,6)-(4,8)$	0.50773
4	14	$(14,1)-(13,2)-(9,4)-(6,6)-(3,8)$	0.50193

Table 2: Code decompositions from [7, Table 5] with their corresponding lower bounds on kernel exponents for the kernels induced by them.

then there exists a binary code decomposition of $\{0,1\}^{\ell}$, *such that*

$$D_{min}^{(i)} \geq d_j \quad \text{where} \;\; k_{j+1} < \ell - i + 1 \leq k_j,$$

$$j \in [m], \;\; i \in [\ell], \;\; k_{m+1} = 0.$$

The next observation about the kernel exponent is an easy consequence of the previous observation.

Observation 2 *If there exists a code decomposition of* $\{0,1\}^{\ell}$ *with the following chain of parameters*

$$(\ell, k_1, d_1) - (\ell, k_2, d_2) - \ldots - (\ell, k_m, d_m),$$

then there exists an ℓ *dimensional binary kernel* $g(\cdot)$ *induced by a binary code decomposition* $\{T_1, \ldots, T_{\ell+1}\}$ *such that*

$$E(g) \geq (1/\ell) \cdot \sum_{i=1}^{m} (k_i - k_{i+1}) \cdot \log_{\ell}(d_i), \tag{51}$$

where $k_{m+1} = 0$.

In [7, Table 5], the author gives a list of code decompositions for $\ell \leq 16$. Using this list, we can construct polarizing non-linear kernels and get lower bounds on their exponent $E(g)$ (In order to do so, we use Observation 2 and Propositions 1 and 2). Table 2 contains a list of code decompositions that give lower bounds on $E(g)$ that are greater than 0.5. At the chain description column of the table, the code length equals ℓ for all the sub-codes, and was omitted from the chain for brevity. Note that the second entry of the table has the exponent of the kernel suggested in [3]. It was proven that this is the best linear binary kernel of dimension 16, and that all the linear kernels of dimension < 16 have exponents ≤ 0.5. The first entry of the table gives a non-linear decomposition resulting in a non linear kernel having a better exponent. In fact, this exponent is even better than all the exponents that were recorded in [3, Table 1]. Furthermore, entries $1, 3$ and 4 achieve the optimal exponent per their dimension as Table 1 indicates. Thus, the exponent value indicated in Table 2 is not just a lower bound, but rather the true exponent. The appendix contains details about the decompositions in Table 2.

5 Conclusions

The notion of code decomposition was used for the design of good binary kernels in the sense of the polar code exponent. Some of the kernels we suggested are proven to achieve the optimal exponent per their dimension. It should be noted that by using non-binary kernels one can get better exponents, as was demonstrated in [5]. There is an essential loss, when using non-binary code decomposition for designing binary kernels. It seems that if we allow the inputs of the kernel to be from different alphabet sizes, we may gain an additional improvement. This interesting idea is further explored in a sequel paper by the authors [8].

Appendix

In this appendix we give details on the decompositions enumerated in Table 2. All of the decompositions are coset decompositions, so we only need to specify the sub-code representatives.

#1)$(16,16,1)-(16,15,2)-(16,11,4)-(16,8,6)-(16,5,8)-(16,1,16)$

The sub-code representatives are $(16,15,2)$ single parity check code, $(16,11,4)$ extended Hamming code, $(16,8,6)$ Nordstrom-Robinson code, $(16,5,8)$ first order Reed-Muller code, $(16,1,16)$ repetition code.

#2)$(16,16,1)-(16,15,2)-(16,11,4)-(16,7,6)-(16,5,8)-(16,1,16)$

The sub-code representatives are $(16,15,2)$ - single parity check code, $(16,11,4)$ - extended Hamming code, $(16,7,6)$ - extended 2-error correcting BCH code, $(16,5,8)$- first-order Reed-Muller code, $(16,1,16)$ - repetition code.

#3)$(15,15,1)-(15,14,2)-(15,10,4)-(15,7,6)-(15,4,8)$

The sub-code representatives are $(15,14,2)$ - single parity check code, $(15,10,4)$ - shortened extended Hamming code, $(15,7,6)$ - shortened Nordstrom-Robinson code, $(15,4,8)$ - shortened first order Reed-Muller code.

#4)$(14,14,1)-(14,13,2)-(14,9,4)-(14,6,6)-(14,3,8)$

The sub-code representatives are $(14,13,2)$ - single parity check code, $(14,9,4)$ - twice shortened extended Hamming code, $(14,6,6)$ - twice shortened Nordstrom-Robinson code, $(14,3,8)$ - twice shortened first order Reed-Muller code.

Explicit Encoding of Decomposition #1

For decomposition #1 we elaborate on the kernel mapping function $g(\cdot):\{0,1\}^{16}\to\{0,1\}^{16}$. To do so, we use Table 3. The third column from the left determines whether the vectors on the second column are all the coset vectors (if they do not form a linear space) or just the basis for the space of coset vectors (if they form a linear space). The fourth and the fifth columns determine the stage of the code decomposition these vectors belong to; the "main code" is decomposed to cosets of the "sub-code" (each coset is generated by adding a different coset vector from the set specified by column 2 to the sub-code). The entry corresponding to indices $9-11$ is taken from [9].

We now describe the encoding process. Let $\mathbf{u}_1^{16}$ be a binary vector. The indices of the vector are partitioned to subsets according to the first column of the table. For each subset the corresponding sub-vector of $\mathbf{u}$ is mapped to a coset vector. The mapping can be arbitrary, but when the coset vectors form a linear space, we usually prefer to multiply the corresponding sub-vector by a generating matrix which rows are the vectors in the "coset vectors" column. To get the value of $g(\mathbf{u})$, we add-up the six coset vectors we got from the last step. Note that using this mapping definition, it is easy to derive the mapping function corresponding to decompositions #3 and #4 as well.

References

[1] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, 2009.

[2] E. Arikan and E. Telatar, "On the rate of channel polarization," Jul. 2008. [Online]. Available: http://arxiv.com/abs/0807.3806

input vector indices	coset vectors	coset vectors form a linear space?	main code	sub-code
1	[0000000000000001]	yes	$(16, 16, 1)$	$(16, 15, 2)$
$2-5$	[0000000100000001] [0000000000010001] [0000000000000101] [0000000000000011]	yes	$(16, 15, 2)$	$(16, 11, 4)$
$6-8$	[0001000100010001] [0000010100000101] [0000000001010101]	yes	$(16, 11, 4)$	$(16, 8, 6)$
$9-11$	[0000000000000000] [0000001101010110] [0001000101001011] [0001001000101110] [0001011100011000] [0000011000110101] [0001010001110010] [0000010101101100]	no	$(16, 8, 6)$	$(16, 5, 8)$
$13-15$	[0101010101010101] [0011001100110011] [0000111100001111] [0000000011111111]	yes	$(16, 5, 8)$	$(16, 1, 16)$
16	[1111111111111111]	yes	$(16, 1, 16)$	-

Table 3: Coset vectors for code decomposition #1.

[3] S. B. Korada, E. Sasoglu, and R. Urbanke, "Polar codes: Characterization of exponent, bounds, and constructions," Jan. 2009. [Online]. Available: http://arxiv.com/abs/0901.0536

[4] R. Mori and T. Tanaka, "Performance and construction of polar codes on symmetric binary-input memoryless channels," in *Proc. IEEE Int. Symp. Information Theory ISIT 2009*, 2009, pp. 1496–1500.

[5] ——, "Non-binary polar codes using reed-solomon codes and algebraic geometry codes," Jul. 2010. [Online]. Available: http://arxiv.org/abs/1007.3661

[6] P.Delsarte, "An algebraic approach to the association schemes of coding theory," *Phillips Research Reports*, 1973, supplement No. 10.

[7] S. Litsyn, *Handbook of Coding Theory.* Eds., Elsevier, The Netherlands, 1998, ch. An Updated Table of the Best Binary Codes Known.

[8] N. Presman, O. Shapira, and S. Litsyn, "Polar codes with mixed kernels," 2011, accepted for 2011 IEEE International Symposium on Information Theory.

[9] A. E. Ashikhmin and S. N. Litsyn, "Fast decoding algorithms for first order reed-muller and related codes," *Designs, Codes and Cryptography*, vol. 7, pp. 187–214, 1996, 10.1007/BF00124511. [Online]. Available: http://dx.doi.org/10.1007/BF00124511